

Precision	Sign		Exponent		Mantissa		Bias
	Number of Bits	Bit Range	Number of Bits	Bit Range	Number of Bits	Bit Range	
Single	1	[31]	8	[30-23]	23	[22-0]	127
Double	1	[63]	11	[62-52]	52	[51-0]	1023

Figure 1A

Floating Point Numbers
Under IEEE Standard 754

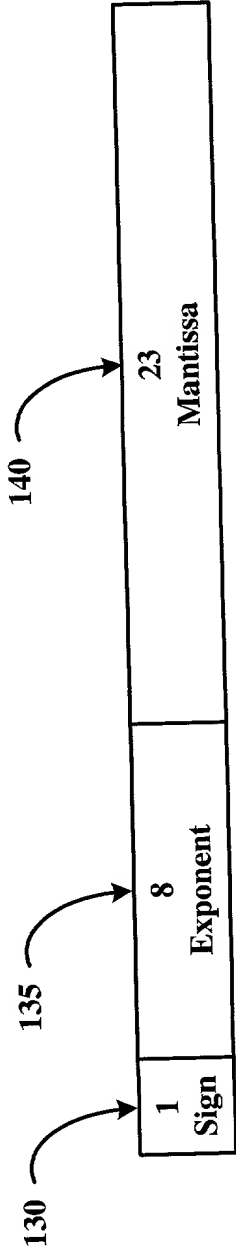
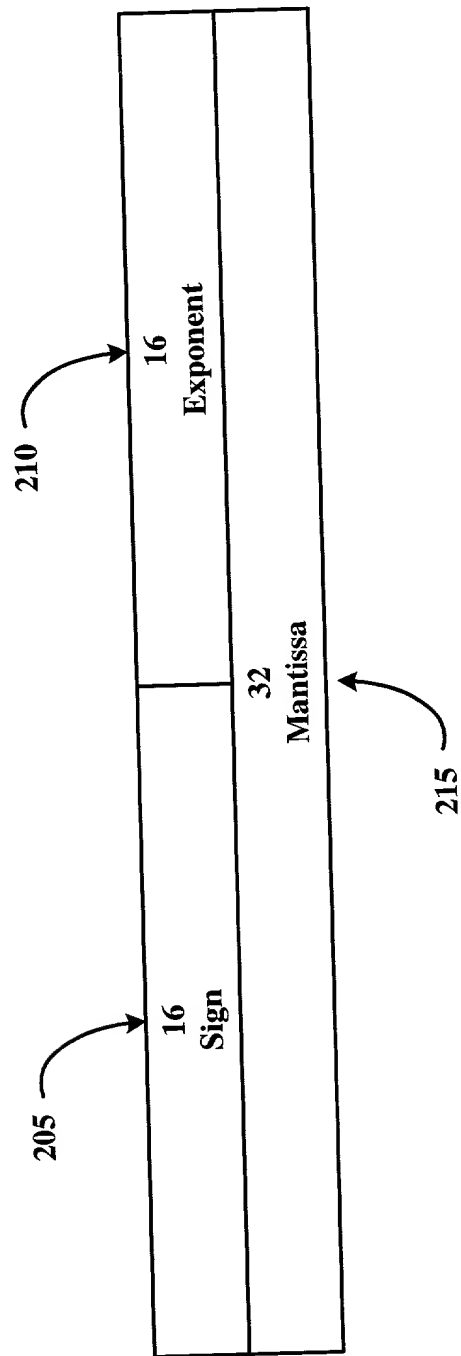


Figure 1B

Floating Point
Single Precision Number



Unpacked
Floating Point Number

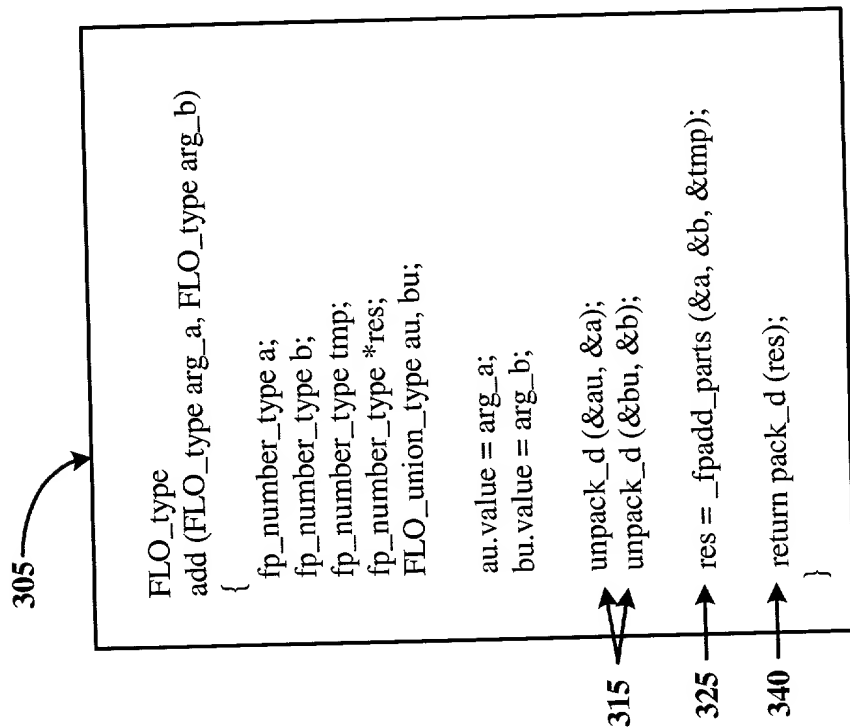


Figure 3A

Conventional Addition Routine

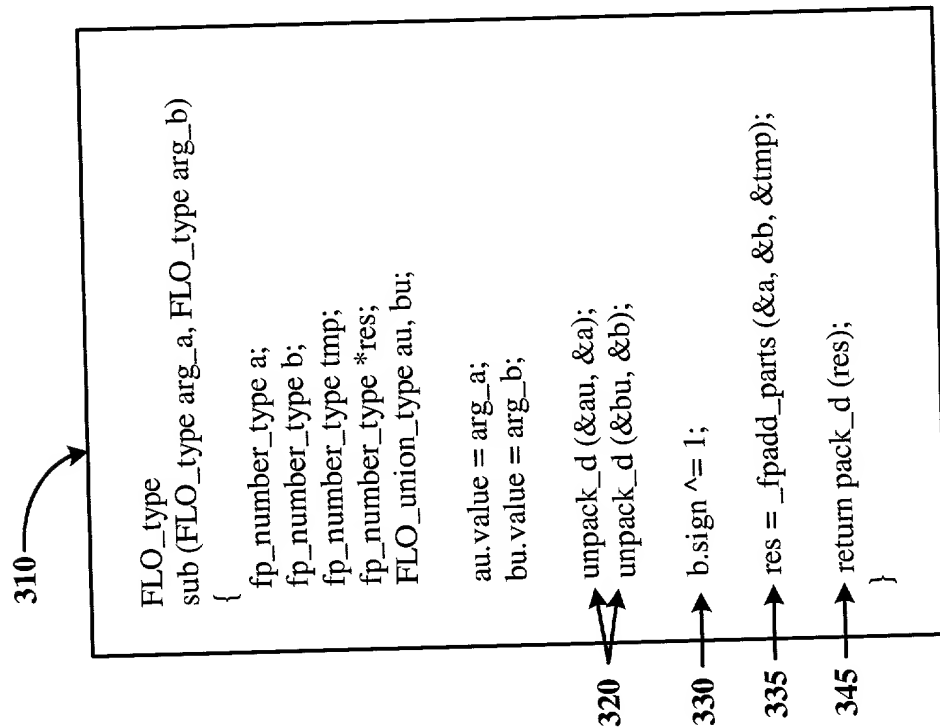


Figure 3B

Conventional Subtraction Routine

405	$T = a \times b \times c$
410	$T_0 = a \times b$
415	$T_1 = T_0 \times c$
420	$T_2 = \text{unpack}(a)$
425	$T_3 = \text{unpack}(b)$
430	$T_4 = \text{unpack_mult}(T_2, T_3)$
435	$T_0 = \text{pack}(T_4)$
440	$T_5 = \text{unpack}(T_0)$
445	$T_6 = \text{unpack}(c)$
450	$T_7 = \text{unpack_mult}(T_5, T_6)$
455	$T_1 = \text{pack}(T_7)$

Figure 4

Calculation Using Conventional
Floating Point Emulation

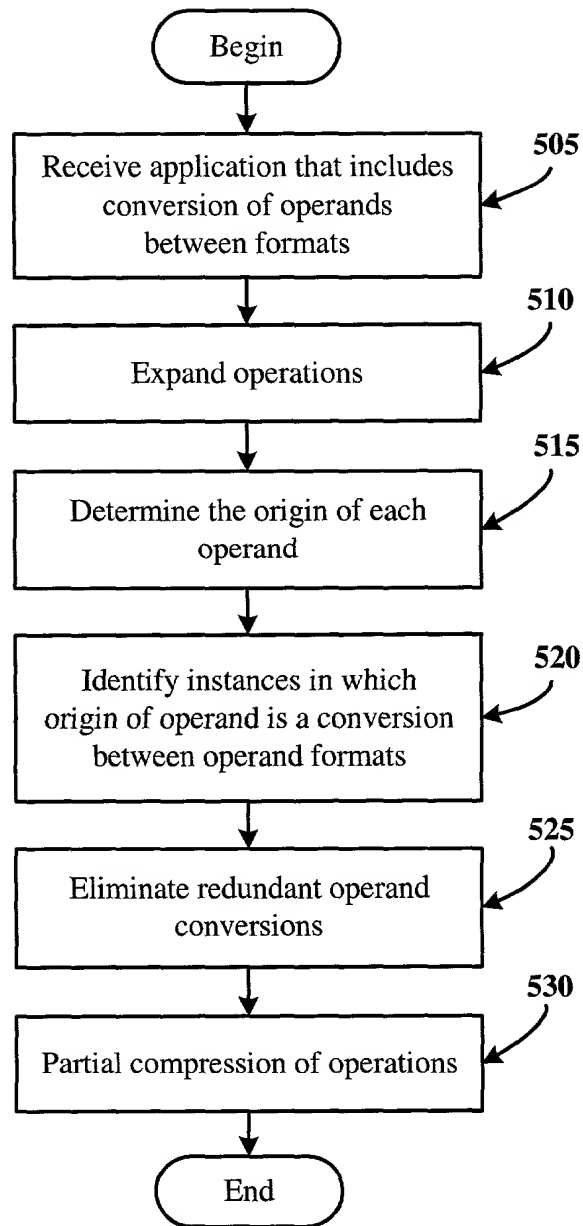


Figure 5

Operand Conversion Optimization

FIG. 6

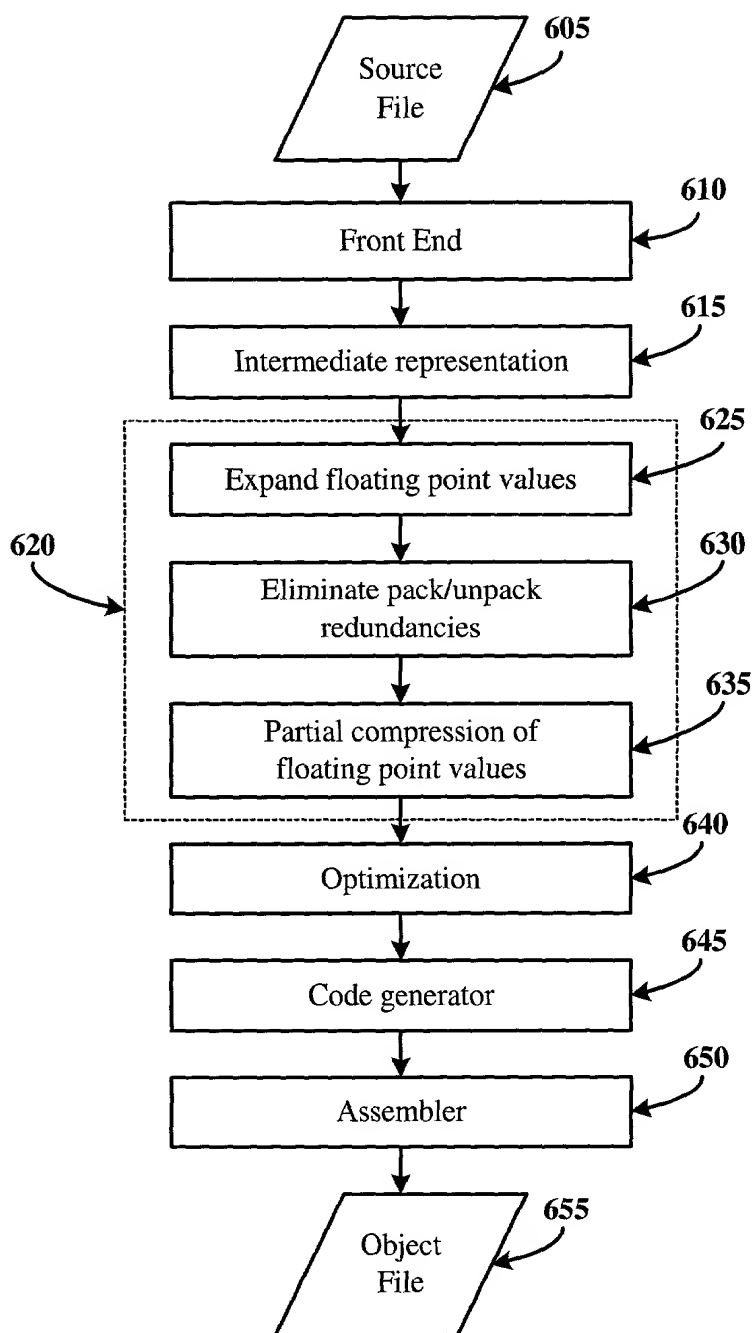


Figure 6

Exemplary System Including
Operand Conversion Optimization

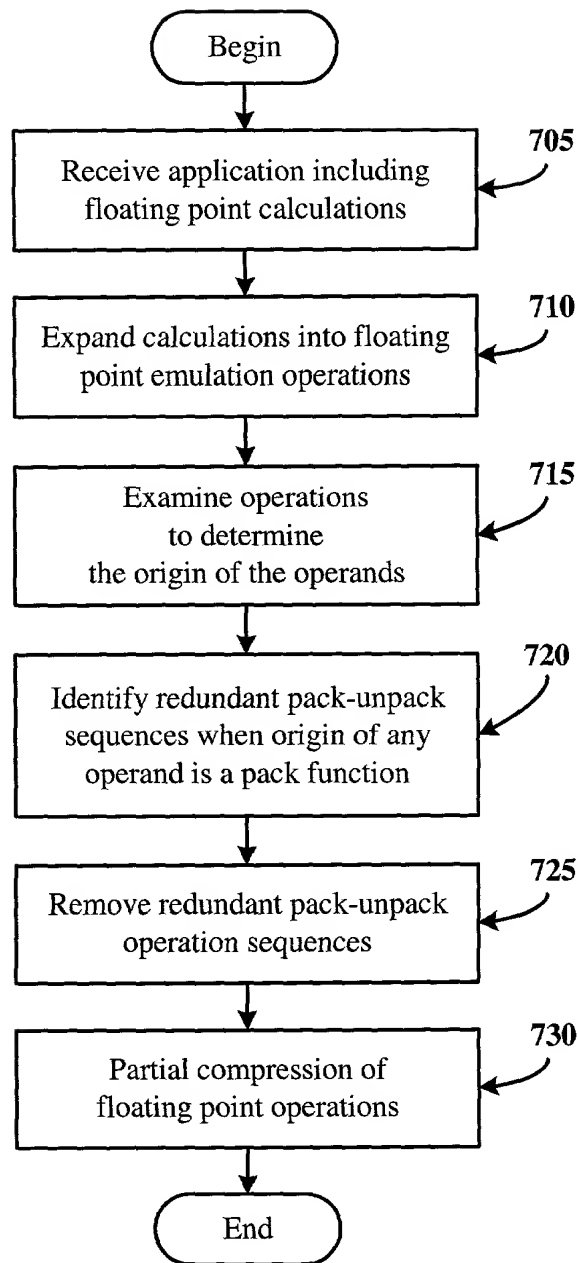


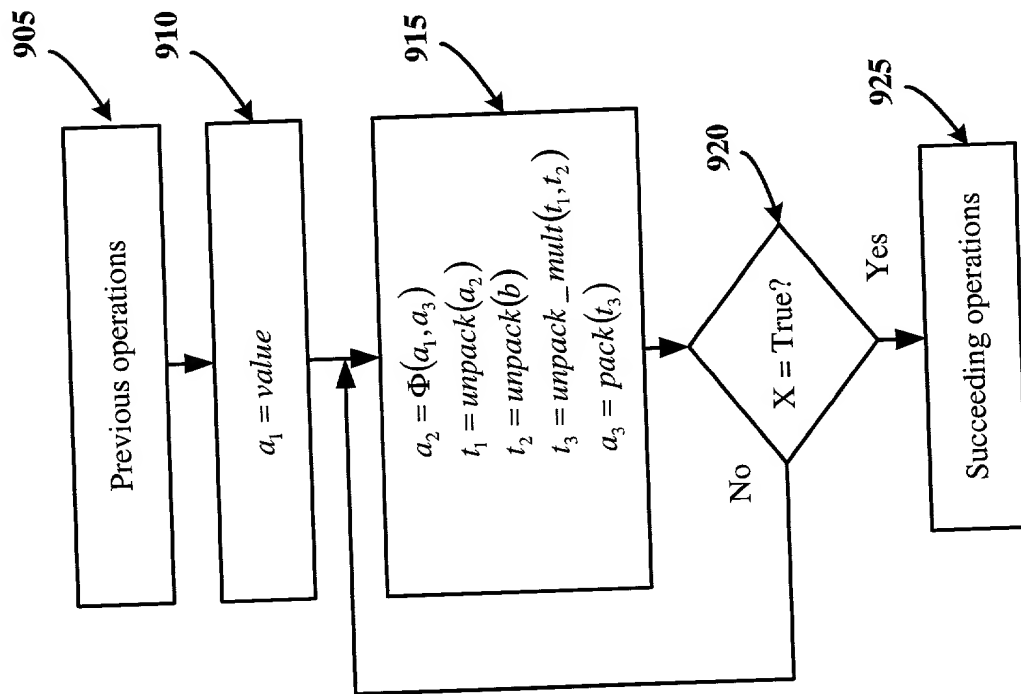
Figure 7

Optimization of
Floating Point Emulation

805	$T = a \times b \times c$
810	$T_0 = a \times b$
815	$T_1 = T_0 \times c$
820	$T_2 = \text{unpack}(a)$
825	$T_3 = \text{unpack}(b)$
830	$T_4 = \text{unpack_mult}(T_2, T_3)$
835	$T_5 = \text{unpack}(c)$
840	$T_6 = \text{unpack_mult}(T_4, T_5)$
845	$T_1 = \text{pack}(T_6)$

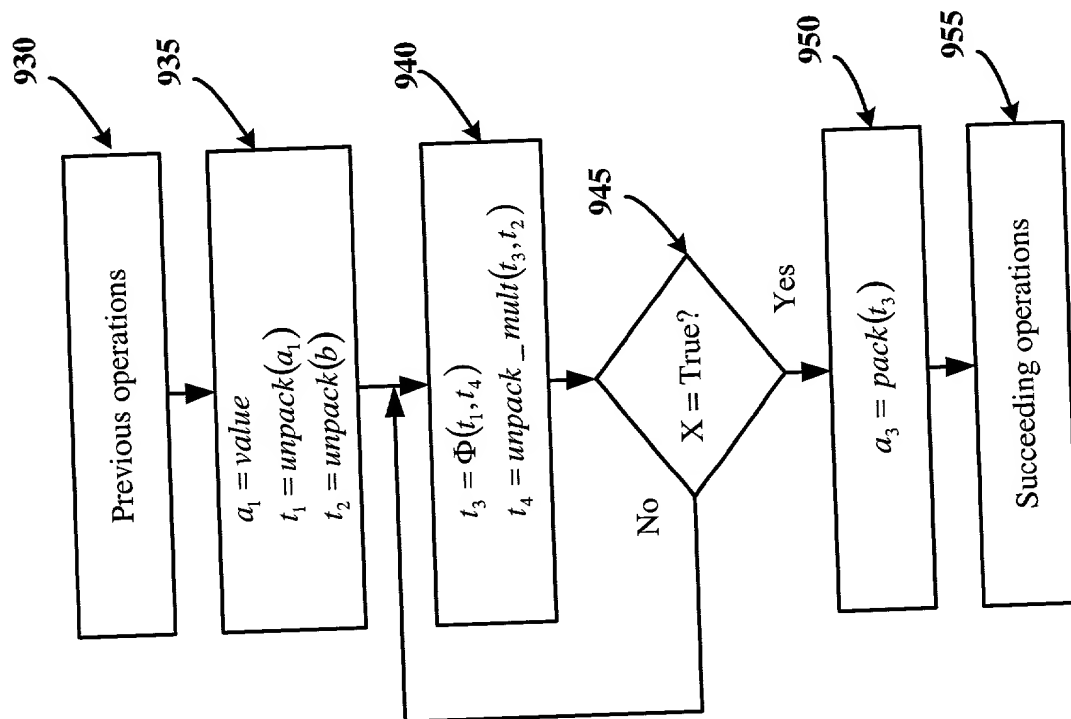
Figure 8

Calculation Using
Emulation Optimization



Loop with
Partial Redundancy

Figure 9A



Optimized
Loop

Figure 9B